NASA Technical Memorandum 84912

The Role of Simulation in the Development and Flight Test of the HiMAT Vehicle

Martha B. Evans and Lawrence J. Schilling

April 1984

National Aeronautics and
Space Administration

NASA Technical Memorandum 84912

The Role of Simulation in the Development and Flight Test of the HiMAT Vehicle

Martha B. Evans and Lawrence J. Schilling
NASA Ames Research Center, Dryden Flight Research Facility, Edwards, California 93523

1984

NASA
National Aeronautics and
Space Administration
Ames Research Center
Dryden Flight Research Facility
Edwards, California 93523

SUMMARY

Real-time simulations have been essential in the flight-test program of the highly maneuverable aircraft technology (HiMAT) remotely piloted research vehicle at NASA Ames Research Center's Dryden Flight Research Facility. The HiMAT project makes extensive use of simulations in design, development, and qualification for flight, pilot training, and flight planning. Four distinct simulations, each with varying amounts of hardware in the loop, were developed for the HiMAT project. The use of simulations in detecting anomalous behavior of the flight software and hardware at the various stages of development, verification, and validation has been the key to flight qualification of the HiMAT vehicle.

NOMENCLATURE

ADC	analog-to-digital converter
ADI	attitude and direction indicator
BCS	backup control system
CASH	computation and simulation of HiMAT
CPU	central processing unit
DAC	digital-to-analog converter
DPM	degraded primary mode
EGT	exhaust gas temperature
EPROM	erasable, programmable, read-only memory
FTMAP	flight-test maneuver autopilot
HiMAT	highly maneuverable aircraft technology
ILS	instrument landing system
I/O	input/output
IPCS	integrated propulsion control system
MCWP	master caution and warning panel
PCM	pulse-coded modulation
PCS	primary control system
RAM	random access memory
RCV	remotely controlled vehicle

RPRV remotely piloted research vehicle

SAE servoactuator electronics

TP proposed touchdown point for a HiMAT vehicle

INTRODUCTION

A major element of the NASA research flight-test program of the highly maneuverable aircraft technology (HiMAT) vehicle was the development and use of a family of complex high-fidelity simulations. These simulations were used for design, development, and qualification of vehicle systems and the planning and training support for all the flight operations. The role of the simulations in this program was extremely important because of the characteristics of the HiMAT vehicle and its operation.

The HiMAT vehicle is a 44-percent-scale version of an envisioned fighter aircraft that has advanced technologies, such as reduced static stability and digital fly-by-wire controls. To significantly increase the maneuverability over current fighters such as the F-15 and the F-16, the vehicle was designed to sustain 8g at Mach 0.9 at 7620.1 m (25,000 ft) and to have good supersonic performance. Because of the unproven technologies used in the design and the ability for the vehicle to sustain high g, the vehicle was flown as a remotely piloted research vehicle (RPRV). The limited number of flights planned for the program, the unstable aircraft configuration, and the remotely piloted vehicle aspects caused the simulations to become essential to the HiMAT program.

Four distinct HiMAT real-time simulations were developed, with varying amounts of flight hardware included. Approximately 2200 hr of real-time simulation were spent prior to the first HiMAT flight. The simplest HiMAT simulation was done originally on a single mainframe computer, whereas the most complex simulation was done on several computers and included the vehicle itself.

At Ames Dryden, simulation and flight of RPRVs overlap. Much of the software developed and used in the simulation is also used in the flight environment. On the other hand, in several of the HiMAT simulations actual flight hardware is used in the simulation environment. The ground-based primary control laws used to fly the HiMAT are designed and developed in the simulation, and are exercised in identical computers in both the simulation and flight environments (ref. 1). Several models besides the control systems are developed in the simulation and then used in flight (for example, the instrument landing system (ILS) and the glideslope). This model is used in the simulation for pilot training, but is also used in the remotely controlled vehicle (RCV) lab as an actual landing aid during HiMAT flights. Where possible, the simulation facility duplicates the computational equipment used in flight. Careful attention was given to model the software and hardware flight systems, including interfaces and time delays, as faithfully as possible. The HiMAT simulations are important to document because they have expanded the knowledge and uses of simulations at Ames Dryden and may have future applications for other projects.

HiMAT DESCRIPTION

The HiMAT research vehicle is turbojet-engine powered and remotely piloted. There are two HiMAT vehicles, one of which is illustrated in figure 1. Both HiMAT vehicles use advanced unproven technologies, such as composite structures, aeroelastic tailoring, reduced static stability, digital fly-by-wire controls, and a digital integrated propulsion control system (IPCS, ref. 2). A cruise (or maneuver) camber to the wing and canard airfoil section is provided by changing leading edges as shown in figure 2. The two vehicles are essentially identical with the exception of their onboard instrumentation.

The 10 HiMAT control surfaces are shown in figure 3. The canards, elevons, and rudders can be driven symmetrically and asymmetrically. The elevators move only symmetrically, and the ailerons (which were locked after the first few flights) move only asymmetrically.

Flight Operations

Major flight operational elements of the HiMAT RPRV system are shown in figure 4. The HiMAT vehicle is carried aloft by a B-52 aircraft and launched near 13,700 m (45,000 ft). The pilot flies HiMAT from a ground-based cockpit that is linked to a set of ground-based computers. Air-data and vehicle-status parameters are downlinked to the ground station by way of telemetry. Pilot commands are interpreted through ground-based primary control-law computers and combined with downlink information to compute surface commands that are uplinked to the vehicle. An onboard backup control-law computer that is capable of flying and landing the vehicle is available. Using cockpit instruments including a glideslope error indicator, television transmission from the vehicle, and calls from the chase pilot, the pilot lands the vehicle on skids on the dry lakebed.

Onboard Computer Systems

Two airborne computers, designated as primary and backup, execute different software (ref. 3) and are both normally active. The onboard computers can be used for backup operation if the ground-based primary control-law computer fails. The backup control system can be controlled internally from onboard sensors, or from discrete commands from either the ground pilot or the airborne controller (the backseat pilot in the chase TF-104G aircraft). The onboard computers also contain other functions, such as an IPCS, uplink and downlink processing, aircraft control by way of ground-based primary control-system laws, failure detection, and intercom control and response.

RPRV Ground Systems

Figure 5 shows the HiMAT ground cockpit. The pilot's inputs to the primary control system consist of a standard three-axis stick and rudder system as well as a throttle and various switches. The pilot's inputs to the backup control system (BCS) are made through a discrete input panel (fig. 6) located on the right side panel of the cockpit and consist of a nine-position joystick, which can command combinations of climb, dive, left, and right (for example, climb to the right), and mode selection switches. The BCS modes available include climb/dive, left-turn/

right-turn, orbit/exit orbit, roll-rate command/attitude command, normal/land, and return to nominal schedule. The pilot also has a switch on the left side panel (fig. 7) that allows him to increase or decrease speed while in backup operation.

The major components of the ground facility are shown schematically in figure 8. Downlink parameters are sent to the telemetry decommutation station and then passed to two front-end/control-law computer pairs (systems A and B) and to cockpit instruments.

The front-end computers receive the input (telemetry downlink) data, select those parameters wanted, and decommutate them for use by the control-law computers. The front-end computers also interpret a set of downlink discretes indicating the health and status of various onboard systems and present the results on a panel referred to as the master caution and warning panel (MCWP). The MCWP is situated remotely from the cockpit and monitored by engineers during flights. The downlink signals coming from the front-end computers, combined with pilot command signals from the cockpit, are inputs to the ground-based control laws.

System A contains the primary control-law computer and also performs air-data calculations and sends this information (Mach number, altitude, vertical velocity, airspeed, dynamic pressure, fuel quantity, and fuel flow) to cockpit instruments. System B contains the laws for a flight-test maneuver autopilot (FTMAP) designed to provide control of the vehicle during selected maneuvers. System B also interprets inputs from and sends outputs to the thumbwheel control box located on the left side panel of the cockpit. The thumbwheel control box is used for real-time inputs to the FTMAP during flights. The system B control-law computer passes ILS/glideslope information to cockpit instruments.

Remotely piloted research vehicles depend on radar information, including vehicle altitude and x-y distances from the radar site, for space positioning. The radar handler receives and decodes radar data. These data are then sent to a digital computer, called the radar computer, which computes ILS/glideslope information. The glideslope and localizer guidance provides error signals on the attitude and direction indicator needles. The guidance was designed to have varying sensitivity that increases as the touchdown point is approached. The radar computer controls a large mapboard and a small x-y plotter that are positioned near the cockpit and are used by the flight-test engineer and pilot for space positioning, navigation, and energy management during flights.

Surface commands from the system A control-law computer and cockpit discretes are sent to an uplink encoder for transmission to the aircraft. Eight 16-bit words are transmitted to the vehicle 53.3 times/sec. Television video is displayed in the cockpit from a forward-looking camera located in the vehicle canopy. It is used only during the approach and landing task.

COMPONENTS OF SIMULATIONS

Four simulations were developed for the HiMAT project. The first simulation, named BASIC, has all components modeled in software. The BASIC simulation is the primary tool for design and development of the control systems and is used for pilot training and flight planning. The second simulation, named VERIFICATION, has the primary control laws resident in computers identical to the ground-based control-law computers used in flight. This simulation is used to conduct verification testing on the primary flight-control system. The third HiMAT simulation is known as CASH (computation and simulation of HiMAT) and has both the primary and the backup control laws resident in computers identical to those in flight. The CASH program is the tool used for verification of the backup control laws and for pilot training, especially for failure mode and effects testing. The fourth simulation is known as the IRON BIRD and includes the vehicle itself in the loop. It is used for full-system validation, dynamic-response tests, limit-cycle tests, failure mode and effects testing, pilot evaluation and training, and complete mission simulation.

All of the HiMAT simulations have at least one background and one real-time loop. A real-time loop is one which is periodically interrupt-driven and executes at high priority, whereas the background loop executes at low priority on a time-available basis. All initialization and non-real-time input/output is done in the background loop or loops.

Appendix A lists functions modeled in the HiMAT simulations. All of the Dryden HiMAT simulations are six-degree-of-freedom simulations based on the following assumptions:

1. Rigid airframe.
2. Earth inertial-reference frame for body-axis equations.
3. Air-mass inertial-reference frame for wind-axis equations.
4. Aircraft symmetric about x-z body plane.
5. Absolute values of angles of attack and sideslip less than 90°.
6. Flat earth.

In developing the family of HiMAT simulations, various simulation components were modeled, some in more than one way. Table 1 lists the more important models used in each of the different simulations. The models are listed either as mathematical or as the actual hardware component. If no computer is listed with the mathematical models, the model resides in the main simulation computer. An understanding of table 1 is critical to understanding how the various simulations are composed. The BASIC simulation contains all the mathematical models that reside in the main simulation computer, whereas most of the models in the IRON BIRD simulation were actual flight hardware. This section describes the more important individual components and how they were simulated. The next major section, HiMAT SIMULATION SYSTEMS, discusses the four simulations and how they were used.

Aerodynamic Model

Actual. - The actual aerodynamic system is the response of the vehicle in flight.

Simulation model. - The original HiMAT aerodynamic data were based on analytical estimations and a very small amount of preliminary wind-tunnel data furnished by Rockwell International, Los Angeles Aircraft Division. These data, along with a set of equations for computing total force and moment coefficients, composed the original aerodynamic model. Rigid aerodynamic data and flexible-to-rigid ratios for both the maneuver-wing and the cruise-wing configurations were included. The flexible-to-rigid ratios indicate how much the aircraft coefficients will change due to aircraft deformation caused by dynamic pressure and loads. This data set was made up of 112 separate arrays (21,500 data values).

This data set was changed based on a minimal verification wind-tunnel test performed at the NASA Ames Research Center. The wind-tunnel model was configured for Mach 0.9, 9144 m (30,000 ft) altitude, and 1g normal acceleration. The actual vehicle was loaded to 8g, and structural deflections were measured. Using these data, estimated structural influence coefficients and flexible-to-rigid ratios were derived. These ratios were used to derive rigid characteristic data from the first wind-tunnel data, and the resulting data became the second aerodynamic data set. The values in this set varied greatly from those in the original set.

It should be noted here that the discrepancies between the first two sets of aerodynamic data caused great concern, since the vehicle was an RPRV and was to fly in an unstable configuration. Having little faith in the fidelity of the aerodynamic model which was used in the simulation, the project office decided to fly first in a stable configuration to get flight-determined data and to confirm or deny the validity of the simulation model. It was imperative that the aerodynamic model be good, because the control laws were designed using that model. The HiMAT aerodynamic model is what makes the simulation a HiMAT model rather than some other vehicle, such as an F-15 aircraft or a space shuttle. If the aerodynamic model were inaccurate, the adequacy of the control systems would be in question.

Given more detailed wind-tunnel testing, one would expect a better correlation between the estimated and flight data. Areas which proved significantly different between estimated wind-tunnel data and flight data included the lateral-directional derivatives and data in the supersonic region. Flight-test data also indicated that flexibility terms did not have a significant effect on the aerodynamic coefficients; therefore, a simplified aerodynamic data set (derived from flight data) and corresponding force and moment coefficient equations were developed. This data set became the third basic aerodynamic set used for simulations and used only 29 data arrays (6770 data values), approximately one-fourth the number used in the full aerodynamic set with flexibility effects.

The simplified third aerodynamic set included a combined coefficient of drag and of lift. Both coefficients were computed from the full aerodynamic set as functions of angle of attack, Mach number, and elevon position, assuming that elevators were at the same position as the elevons. The computation of these two terms in the full aerodynamic coefficient equations incorporated many individual components. This approach required less memory and time to look up tables and compute equations; however, it proved difficult to isolate and adjust individual component inaccuracies.

Primary Control System

Actual. - The primary flight control system (PCS) is one of two independent flight control systems (primary and backup) required by the HiMAT program. The PCS control law is resident in a ground-based digital computer and is designed to fly the vehicle in the relaxed static stability configuration. The longitudinal control law consists of two distinct parts. The first part provides a normal controller (no angle-of-attack or normal acceleration limiters) with a pitch-rate-command augmentation system, and the second part provides angle-of-attack and normal acceleration limiters. This control law incorporates a forward-loop integrator that trims the vehicle longitudinally as long as there is no stick input. Forward-loop integration provides infinite gain to the system but must be disabled for ground checks or the surfaces will integrate to their limits.

Inputs to the longitudinal control law include pilot's stick, normal acceleration, angle of attack, and pitch rate. Mach number and dynamic pressure are used for gain scheduling. In the normal controller, the angle of attack provides improved command response and no stability augmentation, and is used only in supersonic flight. The lateral-directional control law is relatively conventional and provides roll-rate, yaw-rate, and lateral-acceleration feedbacks. Feedback gains are a function of Mach number or dynamic pressure, or both. Pilot input commands are proportional, and the rudder is provided with an aileron-to-rudder interconnect, which is a function of the angle of attack.

Outputs of the PCS consist of surface and throttle commands. A subset of the PCS is the degraded PCS, which commands only the elevons and rudders and adjusts gains through the system to account for the canard, elevator, and aileron surfaces that are locked out. In the PCS mode, the throttle operates using the IPCS in one of the two digital computers onboard the vehicle. The pilot's throttle command consists of both proportional and discrete signals. The stable-configuration longitudinal control laws provided only a normal controller and no pitch-rate feedback augmentation, no forward-loop integration, and no angle-of-attack or normal acceleration limiters. However, these laws did provide a stall inhibitor based on angle of attack. The lateral-directional laws were much the same as for the unstable configuration.

Simulation model. - The simulation of the PCS was made by coding the laws in FORTRAN for the main simulation computer. The actual PCS laws reside in the ground-based control-law computer (which is different from the main simulation computer). The differences between the flight and simulation coding are minimal, and are caused mostly by differences in the computers. The flight computer has most of its routines coded in FORTRAN; however, a few of the routines dealing with interrupts, and uplink and downlink processing are coded in assembly language. These routines were converted to FORTRAN for the BASIC simulation. In all the simulations other than the BASIC simulation, the actual laws are used and are resident in computers identical to those used in flight. The PCS for the unstable vehicle configuration is shown in appendix B.

Backup Control System

Actual. — The backup control system (BCS) is the second of the two independent flight control systems required for the HiMAT program. The BCS control law is resident in one of the two onboard digital computers. The BCS is a full-authority, three-axis, multirate digital controller with stability augmentation functions and mode command functions (ref. 4). Each of seven modes is semiautomatic with the pilot providing direction by way of discrete command inputs. The BCS commands elevons for pitch and roll control and rudders for yaw control, and has an autothrottle for speed modulation.

The BCS was designed to provide well-controlled dynamics throughout the flight envelope, to have the ability to recover from extreme attitudes, and to bring the vehicle to a selected site and effect a successful landing by either a ground-based pilot or an airborne controller (the backseat chase pilot in the TF-104G aircraft). It was designed to provide these features for an unstable vehicle configuration of no more than 10-percent aft mean aerodynamic chord center-of-gravity location. The original HiMAT BCS was developed by Teledyne Ryan Aeronautical for the onboard microprocessor computer, and was programmed entirely in Intel 8080 assembly language.

Simulation model. - The BASIC HiMAT simulation BCS is a FORTRAN implementation of the same flow charts used for the coding of the onboard computer. It provides all of the features that the actual BCS has; however, no attempt was made to model the other functions of the onboard computers, such as the IPCS. The CASH simulation uses computers identical to those used in flight. Because the BCS laws required extensive computational time, they were not included in the VERIFICATION simulation which must meet the 18.75-msec time frame required to simulate the uplink to the onboard computers. Extensive pilot and engineering evaluation was done on the BCS using both the BASIC and the CASH (with actual onboard computers) simulations prior to first flight. Several modifications were made to the BCS based on these evaluations. Other modifications to the BCS were made compatible with the chase aircraft, a TF-104G. Simplified diagrams of the current BCS are shown in figures 9 to 13.

Flight-Test Maneuver Autopilot

Actual. - An FTMAP was developed for HiMAT (ref. 5). It was designed to provide precise, repeatable control of the HiMAT vehicle during selected maneuvers so that a large quantity of high-quality flight data could be obtained in a limited amount of time. The FTMAP performs prescribed maneuvers while maintaining critical flight parameters within close tolerances. The FTMAP operates as an outer-loop control to the primary control system and is located in the ground-based system B control-law computer. When active, the FTMAP replaces normal pilot stick commands, and throttle position and corresponding commands are generated in the FTMAP computer. The pilot retains rudder pedal control to trim sideslip. No FTMAP input was used in the yaw axis.

Simulation model. - The FTMAP laws were designed and run in simulation computers identical to the ground-based flight control-law computers, and only simulations run on those computers had access to FTMAP. The FTMAP laws were very complex and required more computation time than was available in the main simulation computer; therefore, no FTMAP was provided for the BASIC simulation.

Uplink System

Actual. - The uplink system consists of one encoder on the ground and two decoders in the aircraft. The Babcock Encoder Model BCC43A is formatted to send four 16-bit words/frame at a frame rate of 106.6 frames/sec. Two different coded frames are sent alternately, for a total of eight words updated 53.3 times/sec. The airborne portion consists of two receivers, a diversity combiner (ref. 6), and two Babcock BCRD31-B decoders. Each receiver is connected to an antenna (upper and lower). The output of the receiver is fed to the diversity combiner, which electronically mixes the signals and feeds the combined signal to the decoders. Each decoder will accept one frame of data (four 16-bit words). The output of the decoder is passed to the onboard computer.

The surface commands output from the ground-based control laws are converted from engineering units to counts and shifted to the high-order 10 bits of 16-bit words. The throttle command, converted to counts, is multiplexed with a word containing eight packed discretes, and this word is also placed in the high-order 10 bits of a 16-bit word. The remaining 6 bits of the uplink words are hard-wired to cockpit discretes.

Simulation model. - The simulation of the uplink system in the BASIC mathematical model included converting from engineering units to counts, shifting the data words to achieve the same 10-bit system of the actual uplink, and transferring the data arrays from a common block used only in non-control-law routines. The counts were then reshifted and converted to engineering units before being used in the acutator model. This simulation of the uplink system provides realistic discretization effects and timing delays.

For simulations other than BASIC, the uplink data were converted to engineering units, the bits were shifted, and the uplink data arrays were set up in the control-law computer, which is exactly the same as the computer used in flight. The main simulation computer received these arrays and had only to shift the bits and convert the data to engineering units prior to using it.

Downlink System

Actual. - The downlink in the aircraft is a Vector Model MP-600 operating at a rate of 110 kbits/sec. The format is a 10-bit word, 50 words/frame, with 16 subframes. The decommutation station on the ground consists of an EMR 720-bit synchronizer, a model 2731 frame synchronizer, a model 2736 subframe synchronizer, and a model 2748 data distributor.

The downlink data are in the form of a pulse-coded modulation (PCM) stream sent from the HiMAT to the ground receiving station where it is decommutated into recognizable data words in counts. All these data are made available to the front-end computers, which separate those parameters needed by the control laws and those discretes needed by the master caution and warning panel. At the request of the control-law computer, the front-end computer transfers this block of data to the control-law computer.

Simulation model. - The simulation of the downlink included commutating the data (surface positions and state variables) into the same format sent from the vehicle by way of the PCM system. For any but the BASIC simulation, these data were sent to the front-end computers. By structuring the data to be identical to flight PCM data, the front-end and control-law computers in the simulation environment would behave exactly the same as in the flight environment. For the BASIC simulation with no control-law computers, the data had to be decommutated in the main simulation computer.

This simulation of the downlink system provides realistic discretization effects, allowing the control design engineers to solve problems caused by discretization while still in the simulation environment.

Propulsion System

Actual. - The vehicle is powered by a J85-21 afterburning turbojet engine. A digital electronic engine control system in the onboard computer provides engine control and has two modes of operation. During normal mode the engine operates using digitally implemented versions of the normal J85-21 control system; during combat mode the engine operates at maximum rotor speed and uses nozzle modulation to vary the thrust of the engine, which results in quicker thrust response than the normal operation. There is a high-stability feature available to both modes that increases the engine stability margin by reducing the exhaust gas temperature 38° C (100° F) below its normal operating temperature.

Simulation model. - The simulation of the propulsion system was made from a linear, simplified dynamic model of the engine propulsion configuration (ref. 7). This multimode system provided a model of rpm, thrust, fuel flow, exhaust gas temperature (EGT), nozzle area, and ram drag as functions of throttle position, Mach number, and altitude; however, no attempt was made to model details of the IPCS or of the high-frequency components. Normal and combat modes, and normal and high stability are modeled. Following the first flight of the HiMAT, revisions for the engine model were made that included nonlinear rpm values and more complex fuel flow and thrust schedules. Appendix C contains a block diagram of the engine model.

Actuator Model

Actual. - The HiMAT servoactuator system is composed of four types of control actuators. Type A actuators are tandem-redundant and are used for control of the elevons and rudders. Type B actuators are single actuators used for aileron and canard control. Type C actuators are tandem-forced-summing and are used for elevator control. Type D actuators are used for throttle control. A servoactuator electronics (SAE) box contains all the electronics for servovalve drive current and feed-back control of the actuators. In addition, the SAE box provides excitation power and monitoring points for the servoloops.

Simulation model. - The software actuator model used in the HiMAT simulations has a first-order system that is rate-limited and position-limited. The linear response model uses the NASA Langley Research Center local linearization algorithm (ref. 8) to model the response of the transfer function (A/(S + A)). Hysteresis was added to the basic model to more nearly simulate the surface dynamics.

There is a second HiMAT actuator model that is composed of electronic hardware and resides in the same rack (rack B) that houses the onboard computers for the CASH simulation. All inputs, outputs, scale factors, and phasing of the hardware model are identical to those of the real actuators on the vehicle. This hardware model was necessary because the onboard computers (which are used in the CASH simulation) required much faster response than software models allowed.

ILS/Glideslope

Actual. - The pilot can select ILS/glideslope guidance which uses radar data to provide error signals on the attitude and direction indicator (ADI) needles. A glideslope of 2.82° is used for HiMAT. The HiMAT glideslope, shown in figure 5, was designed to have varying sensitivity, depending on the distance of the vehicle from the proposed touchdown point (TP). The glideslope is initiated at a distance of 16,100 m (17,600 yd) from the TP with an off-nominal error of 197 m (215 yd), resulting in full-scale displacement of the error indicator. The sensitivity of the error indication increases linearly until a distance of 914 m (1000 yd) from TP, at which point the sensitivity remains constant with an error of 11 m (12 yd), resulting in full-scale displacement of the error indicator.

The localizer also has variable sensitivity. At 16,100 m (17,600 yd) from the TP, an off-nominal error from the desired ground track results in the full-scale displacement of the ADI vertical needle. The sensitivity of the error indication increases linearly until a distance of 914 m (1000 yd) from the TP, at which point the sensitivity remains constant with an error of 91.4 m (100 yd), resulting in full-scale displacement of the error indicator.

Simulation model. - In the flight environment, the ILS/glideslope routines reside in a digital computer (called the radar computer), not duplicated in the simulation laboratory. To simulate this guidance, the FORTRAN code used in the flight code was duplicated in the main simulation computer. State variables (altitude and x-y distances) are inputs to the simulation model.

Visual Landing Aid

Actual. - Cues to the pilot during landing included the cockpit instruments, ILS/glideslope error indicators, television transmission from the vehicle, calls on the radio from the chase pilot, and space-positioning calls from the flight-test engineer.

Simulation model. - For most of the program, the landing cues for the pilot in a HiMAT simulation included only the instruments, mapboards, and the ILS/glideslope error indicators. Although these are all valid cues, they could not achieve the same effect as the television transmission used in actual flight. During flight, as soon as the pilot can identify the runway, his scan focuses more on the television picture and less on the cockpit instruments. To help alleviate this lack of fidelity in the simulation, a display of the runways on the dry lakebed was developed on a recently purchased Evans and Sutherland Graphics System.

HiMAT SIMULATION SYSTEMS

To provide the necessary capabilities needed by the HiMAT project, four separate HiMAT simulations were developed. The four simulations each have distinct characteristics necessary for the overall development and qualification of flight software. Table 2 summarizes the uses, advantages, and disadvantages of each simulation. As stated previously, table 1 contains a matrix indicating which models (software or hardware) are used in each of the four simulations. Table 3 contains a list of the Ames Dryden simulation hardware available.

BASIC Simulation

The BASIC simulation (fig. 16) is the simplest and most used implementation. In this simulation, all the models are implemented in software. Programmed in FORTRAN IV, this simulation has a frame time of 25 msec. This simulation provides a benign environment for the user by allowing relative ease of program modification and by using the fewest number of computers of any of the HiMAT simulations.

The BASIC HiMAT simulation provided the principal tool for the final design and development of the primary control system (ref. 9). Preliminary control-system models were designed using linear discrete systems analysis. These preliminary models were then placed in the BASIC simulation. Refinements to the preliminary design, including controllability and handling-quality assessments, and initial pilot evaluations were made in this simulation.

The BASIC simulation was also the principal tool for making modifications to the BCS. The procedure for modifying the onboard BCS was to make the proposed changes in the FORTRAN version implemented in the BASIC simulation, provide engineering and pilot evaluations of these changes, and iterate the modifications until acceptable. The modifications were then made to the assembly language code in the onboard system software.

VERIFICATION Simulation

The VERIFICATION simulation is the least complicated HiMAT simulation system that uses the actual ground-based flight control-law code and computers. As such, it is the primary simulation used for verification of flight code. A major drawback of the VERIFICATION simulation for anything other than verification of flight code is that the backup control system is not modeled.

The VERIFICATION simulation (fig. 17) has the primary control laws resident in the control-law computers and runs at an 18.75-msec frame time. The control-law computers execute the control laws and uplink the commands to the main simulation computer. The front-end and control-law computers are identical to those used in flight and the code is transportable between them.

Verification of the flight code is the process which assures that the control system performs exactly as specified, and that the version in the control-law computers performs exactly the same as the version in the simulation. To achieve this verification, programmed ramps of all control-law input parameters are made in both versions of the laws, and all outputs are plotted on strip charts. Both sets of plots must be identical to be verified. Special tests run on the simulation are designed to test for individual changes. Plots, hard copies of displays, or other documentation listing these tests and their results, are kept as permanent records for the project.

CASH Simulation

The CASH simulation (fig. 18) is used extensively for system validation, flight planning, and pilot training — especially for failure-mode training. It incorporates much hardware identical to that used in flight as is shown in table 1. In this simulation both primary and backup control laws are executed in computers identical to those used in flight. The backup control system is verified in this simulation. Without the actual vehicle in the loop, CASH is the best tool for testing flight software; however, it is a very complex system, with 10 computers in the loop, and as such it is not a good tool for design and development. Because of the flight hardware used in this simulation and the resulting fidelity of the interface modeling, much testing was done with the CASH that would otherwise have required use of the IRON BIRD simulation. This resulted in considerable man-hour and dollar savings.

It was necessary to increase the frame rate to minimize transport delays in the interfacing of flight hardware for the CASH simulation. To accomplish this, computation of the vehicle dynamics was moved into an array processor that is interfaced to the main simulation computers. The aerodynamic model, functions of altitude, and gust modeling were also moved to the array processor. The primary real-time loop runs at 9.375 msec with a slower real-time loop at 25 msec.

Command signals are trunked from the cockpit to the system A front-end and control-law computers where they, with the downlink vehicle parameters, are used as inputs to the primary vehicle control laws. Surface commands are generated and output over the hard-wired uplink to the rack-mounted flight computers that process them and command the hardware actuator models. Analog signals representing the resulting surface positions are input to the main simulation computers, where they are converted to floating-point engineering units and passed to the array processor. The vehicle response is computed in the array processor and returned to the main simulation computers that format the downlink parameters and output them to the front-end and control-law computers. This closes the PCS loop. The state variables are also output to the rack-mounted flight computers as simulated sensor signals, which are used as input to the BCS, closing the BCS loop.

A second set of front-end and control-law computers, system B, receives identical downlink data and uses it as input for the FTMAP. The CASH simulation was used for developing and qualifying the original FTMAP, and for mission planning prior to every flight, allowing potential FTMAP problems to be detected. Another application of the simulation was as a diagnostic tool. For difficulties involving the FTMAP encountered in flight, the simulator could often be used for duplication, analysis, and correction of the problem.

The CASH simulation was used extensively for failure-mode training for the pilot and flight-test engineer. Appendix D lists failures that can be evaluated in the CASH simulation. In this simulation, failures can be induced in two ways: by failing the actual hardware or by failing one of the software models. Failures and noise signals can be induced in the hardware actuators and onboard flight computers through sensor inputs located on the front of the rack that houses the onboard computers and the hardware actuators (rack B).

Depending on the fidelity of the model involved, the failures are induced artificially or by actually failing the model in the same way a real failure would occur, and letting the system automatically do the rest. An example of an artificial failure is a generator alert. If this failure is selected, the generator light in the cockpit flashes and continues to flash until the failure is cleared, but nothing really changes in the program. An example of failing the model is an engine failure. When this failure is selected, a flag is set in the engine model, indicating that the engine has flamed out. The flame-out light is set, and the engine model responds with a decay of rpm and an accompanying loss of thrust. When the rpm drops below certain threshold levels, the engine model sets electric and hydraulic failure flags. These flags cause cockpit lights to come on, indicating generator, generator reset, battery on, and bus tie. To restart the engine, the failure must be cleared and the pilot must go through a prescribed sequence of events before the engine model increases rpm and thrust, and the failure lights go out.

IRON BIRD Simulation

Although verification testing may use one or more of the computer systems from the HiMAT simulation, much of it can be accomplished without simulating the dynamics of the vehicle being tested. Validation, however, is a broader task and requires inclusion of the vehicle dynamics. Validation testing seeks to determine if the system, of which the software is only a part, can accomplish the flight requirements (ref. 10).

The IRON BIRD simulation of HiMAT (fig. 19) was developed to perform (1) critical full-system validation of both the primary and backup control systems, (2) limit-cycle tests, and (3) closed-loop failure mode and effects testing (ref. 11). In this simulation, the actual HiMAT vehicle is used and the PCM downlink is hardwired from the vehicle to the RCV laboratory. The uplink command system is hardwired from the RCV laboratory to the vehicle. All vehicle control loops are active. The system is interfaced with the simulation computers located in the simulation laboratory. Surface positions from the vehicle are sent to the simulation computer. Simulated sensor signals are sent to the vehicle, summed with the actual vehicle transducer outputs, and entered in the PCM downlink system. In this simulation, the RCV laboratory and the the vehicle respond as if the vehicle were in true flight, thereby allowing system validation.

The first attempts at an IRON BIRD simulation of the vehicle in an unstable configuration were not successful because of the large transport delays introduced when the system was interfaced with the simulation computers. Several approaches were attempted, including lead compensation on pitch rate and the use of a linear small-perturbation simulation model that allowed reduced frame times. Neither of these attempts provided a solution to the problem of artificial delays, which caused limit cycles. The onboard computer needed pitch-rate feedback at a sample rate of 4.54 msec, but the main simulation computer, which computed the equations of motion, was running at 18.75 msec.

To decrease this delay time, a hybrid simulation was set up having five analog computers perform the airframe dynamics, with aerodynamic data-table look-up, and functions of altitude, engine simulation, and input/output (I/O) performed in the main simulation computer. The ground-based primary control laws were executed in a set of front-end and control-law computers. The resulting hybrid IRON BIRD simulation was successful and was used in several sessions to (1) perform primary and backup control-system dynamic-response tests, PCS limit-cycle tests, and failure mode and effects testing, and (2) check the software and hardware interfaces. In these sessions, the pilot gained experience using actual flight equipment. The inherent capability of the IRON BIRD simulation to interface with the Aeronautical Test Range Facility provided the opportunity to conduct full mission simulations with all personnel on station. Such simulations were performed prior to the first flight and were very valuable in assessing mission timing and control-room procedures. This simulation was not used for regular pilot training and flight planning because it required the vehicle and a large crew, making it very expensive to operate.

The most recent IRON BIRD simulation incorporated two main simulation computers and an array processor that computes the airframe dynamics, including execution of the aerodynamic model. Because the inclusion of the array processor made a fast frame rate possible, the analog computers were no longer necessary and were removed from the simulation. The removal of the analog computers made it easier to set up the simulation and check the software and hardware interfaces, and gave better repeatability.

The first central processing unit (CPU) of the IRON BIRD simulation is driven by an external interrupt synchronized to the onboard computer, with a frame time of 4.54 msec, and performs only time-critical I/O to the vehicle, and interfaces with the array processor. The second CPU is driven by an internal interrupt with a frame time of 25 msec, and performs most of the I/O, gust modeling, and engine modeling.

CONCLUDING REMARKS

The highly maneuverable aircraft technology (HiMAT) remotely piloted research vehicle (RPRV) uses sophisticated and complex real-time simulations for the development and flight testing of the HiMAT system. Four versions of the HiMAT simulations were developed and used on a regular basis for control-system design and development, failure-mode detection and effects testing, flight planning, and pilot training. Because HiMAT includes the most complex simulations developed at Ames Dryden up to this time, the HiMAT simulation family has been considered as Ames Dryden's state-of-the-art simulations for the past several years.

Each of the four HiMAT simulations has an essential role in the development of the HiMAT system. The BASIC simulation is totally modeled in software, is resident in only the main simulation computers, and is the primary tool used for the design and development of both the primary and backup control systems. In the VERIFICATION simulation, the primary control laws are resident in computers identical to those used in flight. This simulation is used for verification of the primary flight control laws. The computation and simulation of HiMAT (CASH) has both the primary and backup control laws resident in computers identical to those used in flight, and also has high-fidelity hardware actuator models. This system is used for backup control-system verification, flight planning, and pilot training — especially for failure-mode training. The HiMAT IRON BIRD simulation places the vehicle in the loop, and is hard-wired to the remotely controlled vehicle (RCV) laboratory. This simulation uses all the actual flight computers (onboard and ground-based), the vehicle actuators, and the uplink and downlink systems, which incorporate all interfaces between the RCV laboratory and the vehicle. All vehicle control loops are active. The main simulation computer executes the equations of motion, and engine and gust modeling. This simulation is used for full-system validation, dynamic-response tests for both primary and backup control-system flight modes, limit-cycle tests, closed-loop failure mode and effects testing, pilot evaluation and training, and complete mission simulation.

The complexity of the HiMAT system required the use of extensive and varied simulation work. Simulation has been an integral part of the HiMAT program and has been critical in the development of the control systems and in system validation.

Ames Research Center
Dryden Flight Research Facility
National Aeronautics and Space Administration
Edwards, California 93523, May 23, 1983

APPENDIX A — FUNCTIONS MODELED IN HiMAT SIMULATIONS

Aerodynamic model
Primary control system (PCS)
    Control laws
    Control pulses and flutter sequence
    Windup turn guidance
    ILS/glideslope guidance
    Maneuver autopilot
    Closed-loop preflight
Backup control system (BCS)
Uplink system
Downlink system
Flight-test maneuver autopilot (FTMAP)
Propulsion model
Software/hardware actuators
ILS/glideslope
Functions of altitude
Changes in inertia and center-of-gravity (c.g.) shift due to fuel consumption
Autotrim
Six-degree-of-freedom equations of motion
Hinge moments
Launch dynamics
Winds
Gusts
Time delay up to five frames
Position error correction
Variable frame times
Ground plane
Direct mode (no PCS) for testing stick and rudders before launch
Selection of runway
Automatic scale for mapboard, small x-y plotter, and strip charts
Failure-mode training
Thumbwheel switch box used for input to preflight, windup turn, guidance,
    and maneuver autopilot laws
Test ramps of all inputs to PCS
Full CRT displays of most parameters in simulation, updated while in real time.
    Hard copies are available, including an automatic copy when the system goes
    to "HOLD."

APPENDIX B — PCS FOR UNSTABLE VEHICLE CONFIGURATION

Four block diagrams of the ground-based primary control laws for HiMAT are presented (figs. B-1 to B-4). The following is a list of acronyms and symbols used in these diagrams.

ALPHA	angle of attack
A_n	normal acceleration
A_y	lateral acceleration
CLIM	control limit
DAL	left-aileron command
DAP	lateral-stick input
DAR	right-aileron command
DCSC	symmetric canard command
DE	elevator position
DEC	elevator command
DEP	longitudinal-stick input
DPM	degraded primary mode
DR	rudder position
DRC	rudder command
DRP	rudder input
DSBC	speed brake command
DVAC	asymmetric-elevon command
DVL	left-elevon surface position
DVLC	left-elevon command
DVR	right-elevon surface position
DVRC	right-elevon command
DVSC	symmetric-elevon command
DW1B1	backup operation
DW5B5	locked for launch
HIPASS	high-pass filter
I/O	input/output
INTFLAG	logical flag for disabling integrator
K_α	alpha gain
KALM	alpha limit
KA_Q	pitch-rate gain
KARI	aileron-rudder interconnect
KGLM	g limit
KG_{N_Z}	normal-acceleration gain
KG_Q	pitch-rate gain
K_{L_α}	alpha gain
KN_α	alpha gain
KN_D	longitudinal-stick gearing
KN_{N_Z}	normal-acceleration gain
KN_Q	pitch-rate gain
KN_T	total-controller gain
KPMCP	pilot-selectable gain
KR_D	lateral-stick gearing
KRMCP	pilot-selectable gain
KR_P	roll-rate gain
KSB	speed-brake gain
KY_{A_Y}	lateral-acceleration gain

KY_D	rudder-pedal gearing
KYMCP	pilot-selectable gain
KY_R	yaw gain
LEAD-LAG	lead-lag filter
LDEP	launch logic
LOPASS	low-pass filter
M	Mach number
P	roll rate
PAC	aileron pulse
PAO1, PAO2, PAO3, PAO4	filter in negative alpha limiter path
PCS	primary control system
PCSC	symmetric-canard pulse
PEC	elevator pulse
PGO1, PGO2	filter in g limiter path
PLO1	filter in alpha limiter path
PNO1, PNO2, PNO3, PNO4	filter in normal controller path
PRSC	symmetric-rudder pulse
PVAC	asymmetric-elevon pulse
PVSC	symmetric-elevon pulse
Q	pitch rate
QBAR	dynamic pressure
R	yaw rate
RO1	filter in roll axis
SBIN	speed brake in
SBOT	speed brake out
THRC	throttle command
THRP	throttle input
YO1, YO2, YO3	filter in yaw axis

APPENDIX C — ENGINE BLOCK DIAGRAM OF THE HIMAT J-85 VEHICLE

Figure C-1 is a block diagram of the propulsion model. Inputs to this model consist of Mach number, altitude, throttle position, and a discrete (the output of the on-off switch). The value of PLAC varies from 0° and 120°. The output of the on-off switch is used to select either a normal or a combat mode. The primary outputs of the simulation are engine rotor speed, inlet drag, gross thrust, normal fuel flow, exhaust gas temperature, and nozzle area. The following is a list of acronyms and symbols used in this diagram.

AB	afterburner	M	Mach number
EGT	exhaust gas temperature	OMGE	rotational velocity of engine
FD	inlet drag	PLA	power lever angle
FG	engine thrust	WFABO	normal fuel flow
H	altitude		

Functions for the HiMAT Propulsion Model

F1, F15	Determines airflow for ram drag.
F2, F5, F6, F7, F24	Calculates gross thrust.
F3, F8, F9, F10, F25	Calculates main engine and afterburner fuel flows.
F4, F18, F19	Used in calculating afterburner fuel flows, gross thrust, and nozzle area.
F11, F12, F13	Calculates throttle position and rates (dynamic effects).
F14, F16, F17	Used in nozzle rate and dynamic effects for different engine modes.
F20	Calculates engine rotor speed.
F21	Calculates ram drag.
F22, F23	Calculates engine temperature for different engine modes.
F26, F27, F28 F29	Calculates nozzle area for engine temperature control, normal and combat modes, and afterburner.

APPENDIX D — FAILURES THAT CAN BE EVALUATED IN CASH SIMULATION

Electrical systems
- Generator alert
- Generator fail
- Battery bus fail
- Generator bus fail
- Bus split

Engine failures
- IPCS sensors
- Engine fire/overheat
- Main burner flame-out/shutdown
- Throttle ampere reset
- Fuel low

Sensor failures
- Primary air data
- Backup air data
- Frozen or ramp angle-of-attack sensor

Ground computers
- System A front-end computer
- System A control-law computer
- System B front-end computer
- System B control-law computer
- Radar computer

Uplink system
- Bad signal strength
- Bad data accepted (receiver 1)

Downlink system
- Loss of downlink
- Backup discrete fail

Ground cockpit
- Cockpit power loss
- Instrument failure
- Stick signal incorrectly compared

Gear deploy failure

Control surface failures
- Primary hydraulic
- Backup hydraulic
- Simplex actuator
- Secondary loop elevon rudder

Airborne computer
- Primary computer
- Backup computer

Miscellaneous
- Backup accelerometer
- Backup rate gyro
- Battery not armed
- Backup-computer real-time clock

REFERENCES

1. Smith, John P.; Schilling, Lawrence J.; and Wagner, Charles A.: Simulation at the Dryden Flight Research Facility. Paper presented at SES/SFTE Simulation-Aircraft Test and Evaluation Symposium, Naval Air Test Center, Patuxent River, Md., Mar. 16-17, 1982.

2. Arnaiz, Henry H.; and Loschke, Paul C.: Current Overview of the Joint NASA/USAF HiMAT program. Paper presented at Tactical Aircraft Research and Technology Conference, Hampton, Va., Oct. 1980.

3. Myers, Albert F.; Earls, Michael R.; and Callizo, Larry A.: HiMAT Onboard Flight Computer System Architecture and Qualification. AIAA Paper 81-2107, Oct. 1981.

4. Kempel, Robert W.: Flight Experience with a Backup Flight Control System for the HiMAT Research Vehicle. AIAA Paper 82-1541, Aug. 1982.

5. Duke, E. L.; Jones, F. P.; and Roncoli, R. B.: Development of a Flight Test Maneuver Autopilot for a Highly Maneuverable Aircraft. AIAA Paper 83-0061, Jan. 1983.

6. Harney, Paul F.: Diversity Techniques for Omnidirectional Telemetry Coverage of the HiMAT Research Vehicle. Paper presented at the 1981 International Telemetry Conference, San Diego, Calif., Oct. 1981.

7. Bayati, J. E.; and Brown, L. E.: Simplified Dynamic Model of the HiMAT RPRV Propulsion System. Rockwell International Document DAR 12500-03, Nov. 1975.

8. Barker, Lawrence E.; Bowles, Roland L.; and Williams, Louise H.: Development and Application of a Local Linearization Algorithm for the Integration of Quaternion Rate Equations in Real-Time Flight Simulation Problems. NASA TN D-7347, 1973.

9. Petersen, Kevin L.: Flight Control Systems Development of Highly Maneuverable Aircraft Technology (HiMAT) Vehicle. AIAA Paper 79-1789, Aug. 1979.

10. Myers, Albert F.; and Sheets, Spencer G.: Qualification of HiMAT Flight Systems. Paper presented at the 7th Annual Symposium of the Association for Unmanned Vehicle Systems, Dayton, Ohio, June 1980.

11. Myers, Albert F.: Simulation Use in the Development and Validation of HiMAT Flight Software. Paper presented at AGARD 28th Guidance and Control Panel Symposium, May 1979.

Table 1 Important HiMAT simulation models for various configurations

Components	Simulations			
	BASIC	VERIFICATION	CASH	IRON BIRD
Aerodynamic	Mathematical	Mathematical	Mathematical	Mathematical
PCS	Mathematical	Identical to ground-based flight computer	Identical to ground-based flight computer	Ground-based flight computer
BCS	Mathematical	None	Identical to onboard flight computer	Onboard flight computer
Uplink	Mathematical	Identical to flight hardware	Identical to flight hardware	Flight hardware
Downlink	Mathematical	Hardware simulating flight hardware	Hardware simulating flight hardware	Flight hardware
FTMAP	None	Identical to ground-based flight computer	Identical to ground-based flight computer	Ground-based flight computer
Propulsion	Mathematical	Mathematical	Mathematical	Mathematical
Actuator	Mathematical	Mathematical	Electronic hardware model	Vehicle hardware
ILS/glideslope	Mathematical	Mathematical	Mathematical	Flight computer

Table 2 Real-time simulation configuration summary

Configuration	Advantages	Disadvantages
BASIC	Best design evaluation tool. Least complicated system to use.	Compute-time requirements are borderline for real-time operation. BCS is optimistic. Perfect sensors. High resolution. No FTMAP.
VERIFICATION	Best PCS evaluation tool. Minimum system complexity using simulation-facility Varian flight computers.	No BCS operation.
CASH	Best BCS evaluation tool. Optimum model of flight configuration with no vehicle impact.	Not good design tool. Complex system.
IRON BIRD	Maximum use of actual flight hardware. Best flight-system validation configuration.	Complex system. Requires much dedicated hardware and personnel.

Table 3 HiMAT simulation hardware

Hardware	Description
1 Cyber 73-28 Computer[a]	
2 Mod Comp Classic 7870 Minicomputers	1 megabyte of local memory each 128 kilobytes of shared memory 32 ADCs, 64 DACs, 128 input/96 output discretes each
1 Floating Point Systems AP-120B Array Processor	131,584 (128K) 38-bit words of data memory 4112 (4K) 64-bit words of instruction memory
2 Varian V-73 Computers	64 kilobytes of memory each 16 ADCs, 16 DACs, 64 input/64 output discretes each
2 Varian V-77 Computers	64 kilobytes of memory each 16 input/16 output discretes each
1 Varian V-72 Computer (RCV lab only)	64 kilobytes of memory
1 Evans and Sutherland Graphics with PDP11/44 Host Computer	256 kilobytes of memory 16 ADCs (only 8 are available outside of the picture system) IEEE 488 interface bus between Evans and Sutherland Graphics and Varian/Mod Comp computer
2 Onboard microcomputers specifically designed and built for HiMAT, based on INTEL 8080A microprocessor	Maximum 22-kilobyte EPROM Maximum 1-kilobyte RAM
10 Hardware actuator models	
1 General-purpose cockpit station	
2 HiMAT vehicles (used during IRON BIRD)	
Miscellaneous equipment including large mapboard, x-y plotter, strip charts, and patch boards.	

[a]Replaced by Mod Comp Classic 7870 Mini-Computers in 1981.

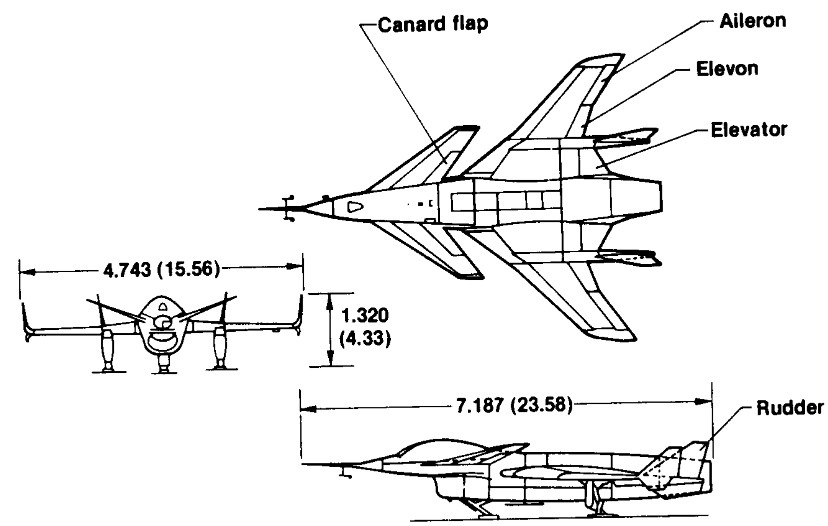


Figure 1. Three-view drawing of HiMAT vehicle. Dimensios in meters (ft).

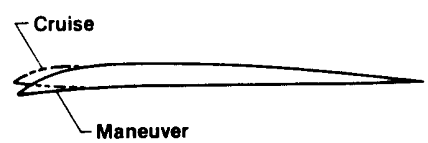


Figure 2. Cruise and maneuver camber leading edge for wing and canard airfoil section (inter-changeable between flights).

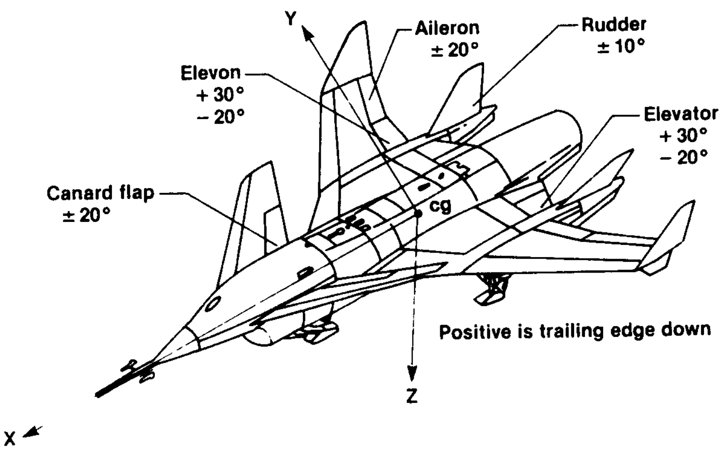


Figure 3. HiMAT vehicle control surfaces.

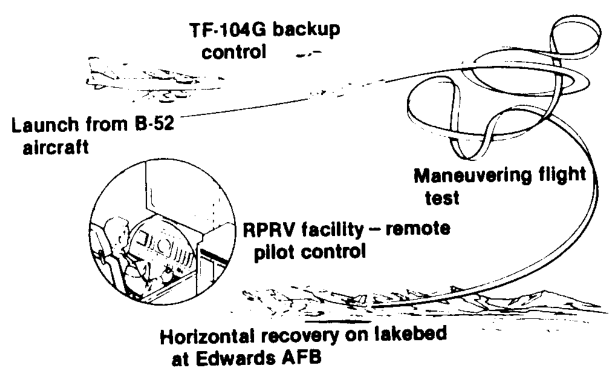
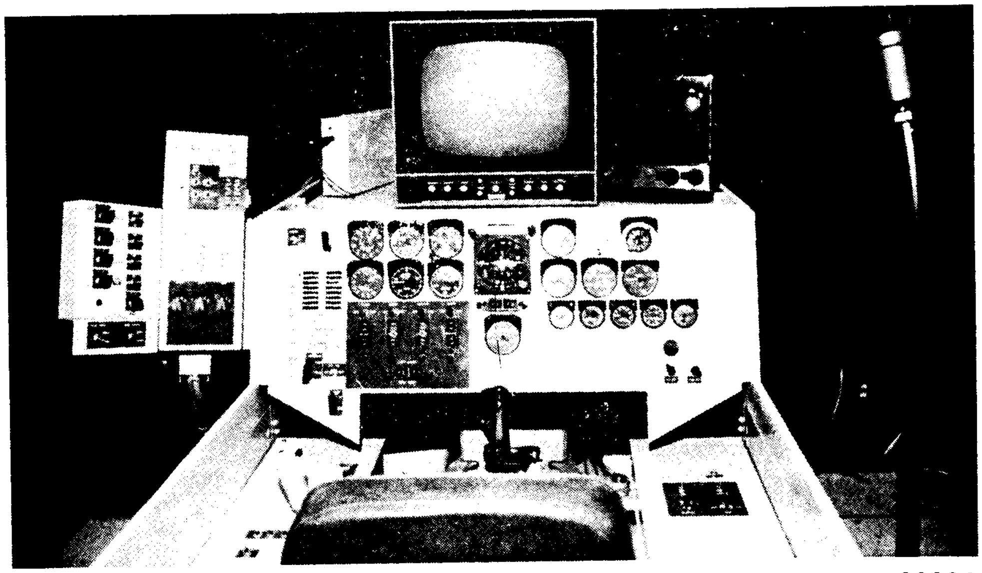
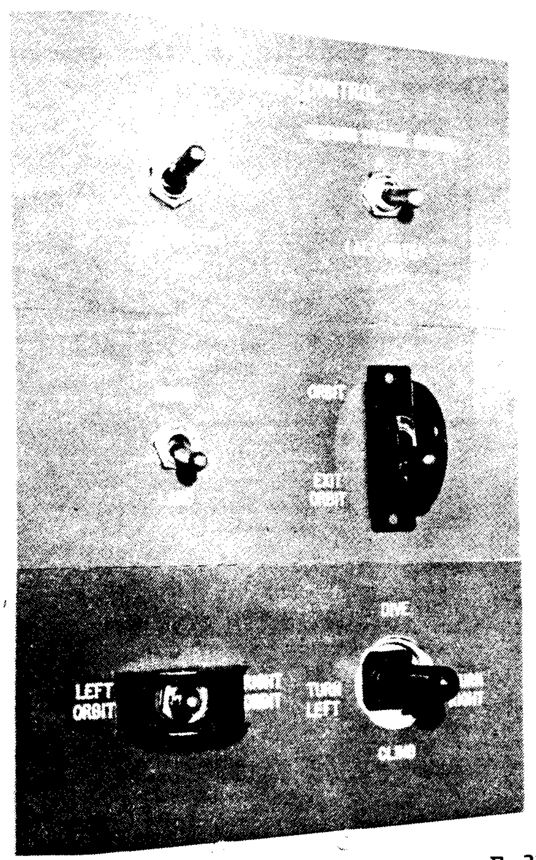


Figure 4. HiMAT operational concept.



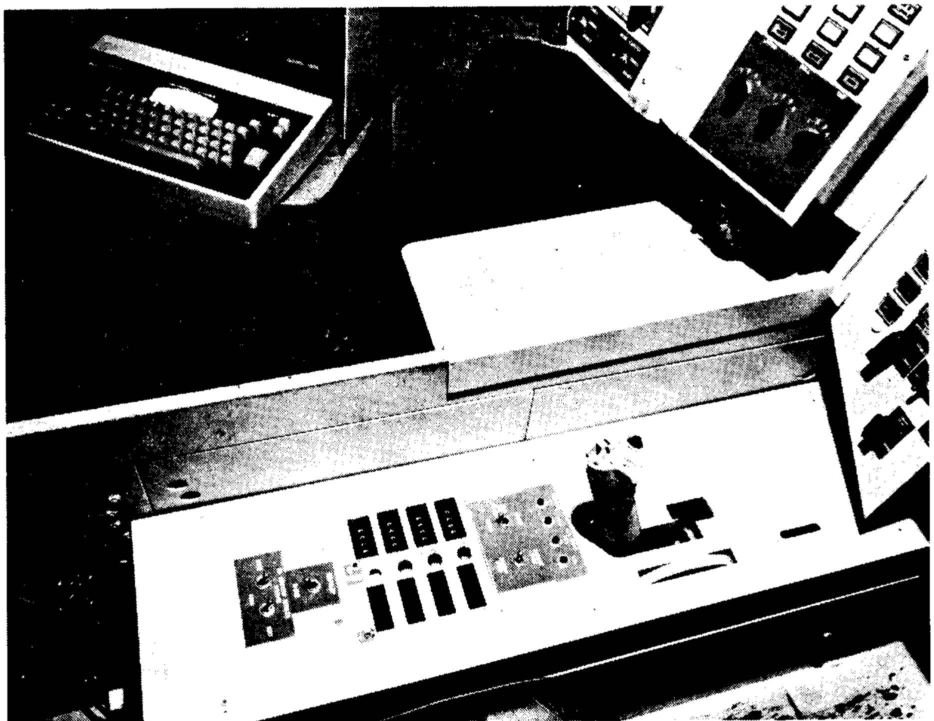
ECN 10108

Figure 5. HiMAT RPRV ground cockpit.



E 37250

Figure 6. Right console in HiMAT ground-based RPRV cockpit.



E 37161

Figure 7. Left console in HiMAT ground-based RPRV cockpit.

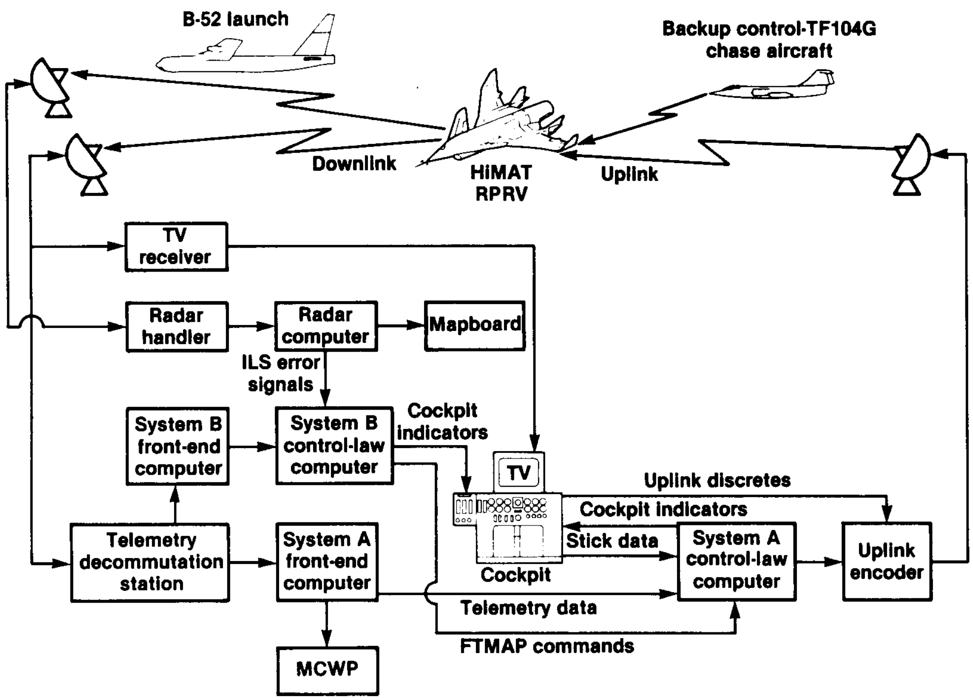


Figure 8. HiMAT RPRV control system.

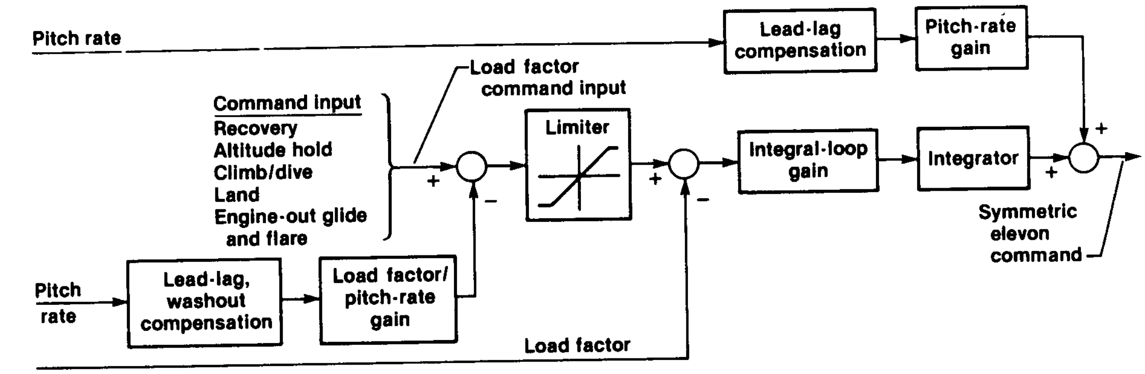


Figure 9. BCS longitudinal stabilization control-law and command inputs.

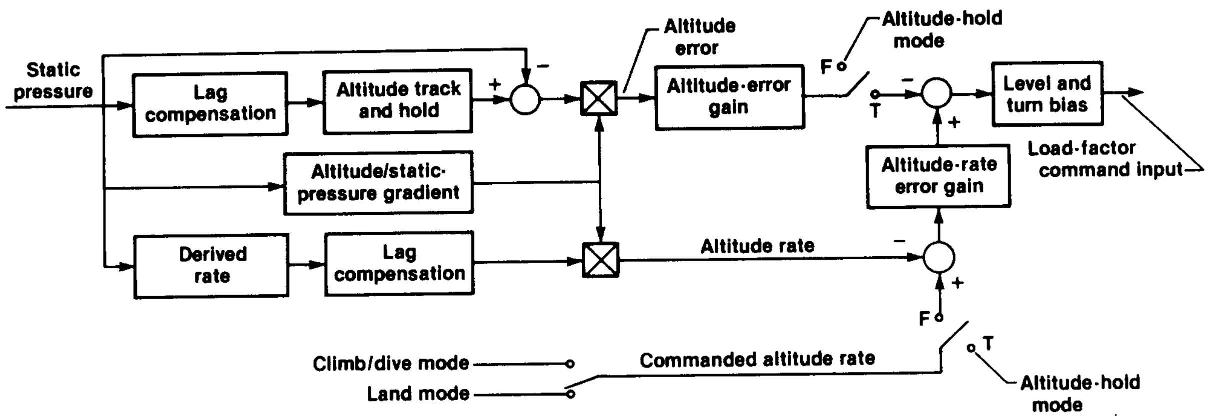


Figure 10. BCS longitudinal recovery-mode command loop.

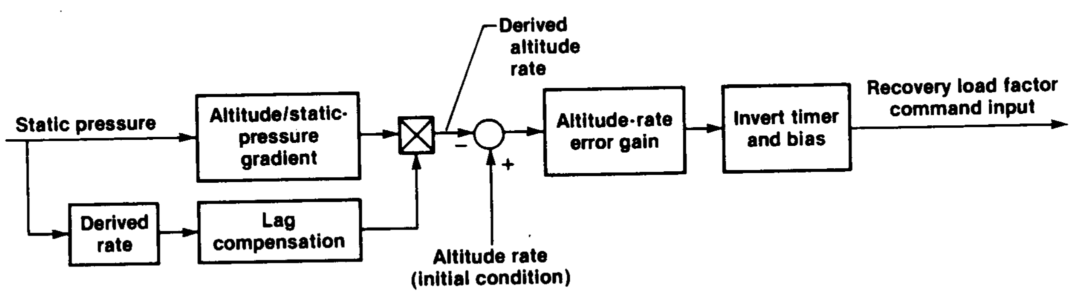


Figure 11. BCS longitudinal command modes.

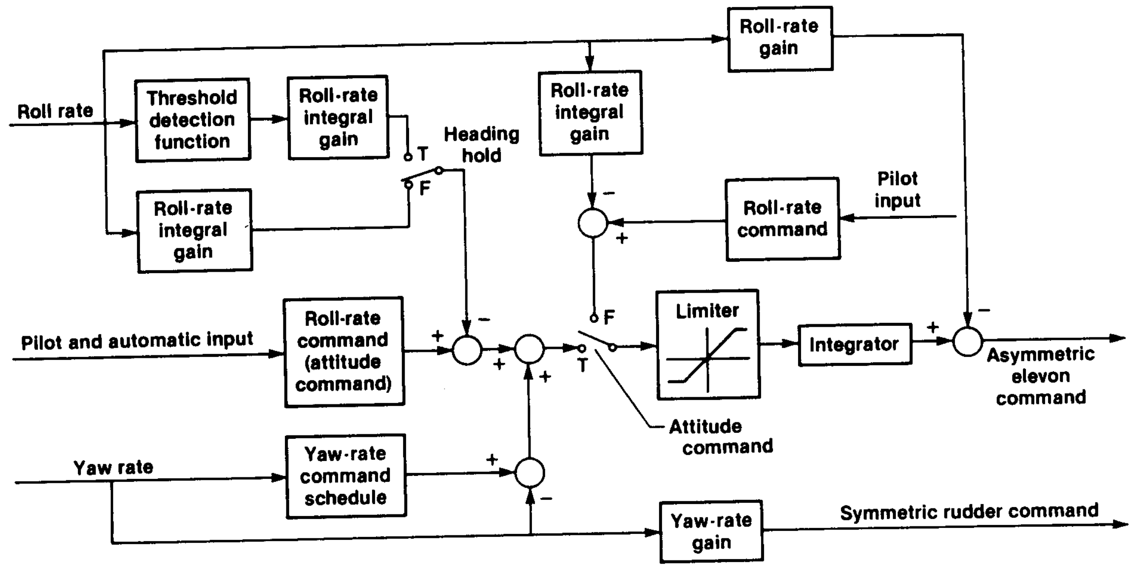


Figure 12. BCS lateral-directional control laws.

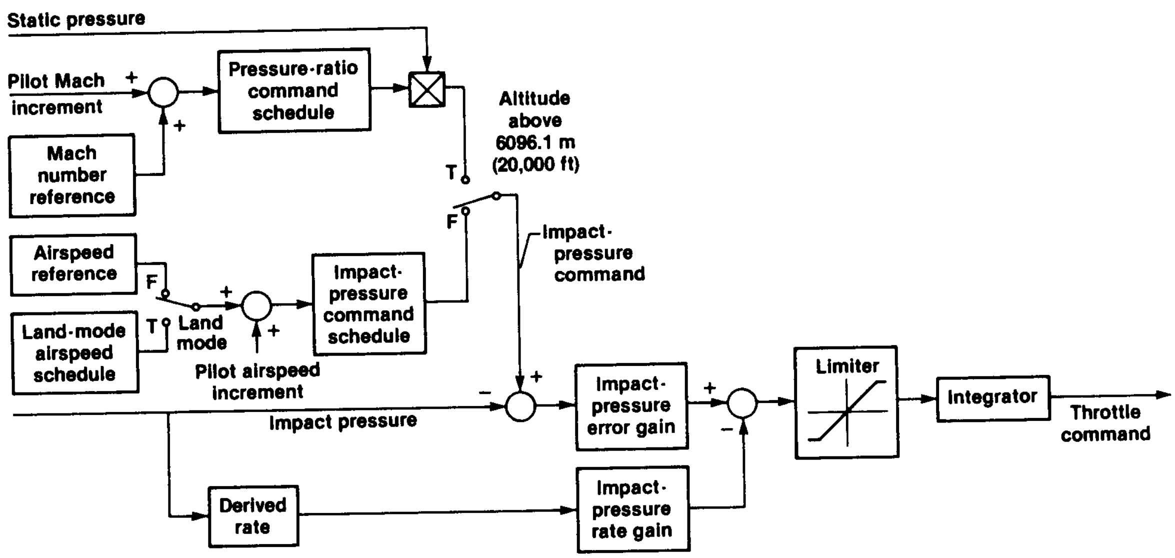


Figure 13. BCS throttle control law.

	Decoder 1 Address		11	12	13	14	15	16
Word 1	Left elevon (Bits 1-10)		Engine operation on/off	Ignitor on/off integrators on/off	Gear down/off	Combat/ normal Return to nominal/off	Engine stability high/ normal	Nozzle override/ normal
Word 2	Right elevon (Bits 1-10)		Climb/off	Descend/ off	Bank right/off	Bank left/off	Speed increase/ off	Speed decrease/ off
Word 3	Rudders (Bits 1-10)		Landing/ standby	Exit orbit/ orbit	Reset bus tie/off	Backup select/ mode	Smoke generate on/off	Rate/ attitude command
Word 4	Drag modulation (Bits 1-9)	Parity (Bit 10) chosen for odd parity	Receiver reset reset/off	Orbit direction left/right	Decoder discrete select/off	Discrete select 1 and 2	Gyro erect/off	Reset generator/ off

	Decoder 2 Address			11	12	13	14	15	16
Word 1	Elevators (Bits 1-10)								
Word 2	Right aileron (Bits 1-10)				Bits	11-16	Same as above		
Word 3	Right canard (Bits 1-10)								
Word 4	Throttle/computer discretes (Bits 1-8)	(Bit 9) Throttle/ discrete	Parity (Bit 10) chosen for odd parity						

1	2	3	4	5	6	7	8
Throttle reset reset/ off	Battery on/off	Bendix status normal/ fail	Onboard pitch rate disengage/ engage	Canard symmetric/ differential	DPM engage/ disengage	Backup select/ normal mode	Test mode active/off

Figure 14. Uplink format.

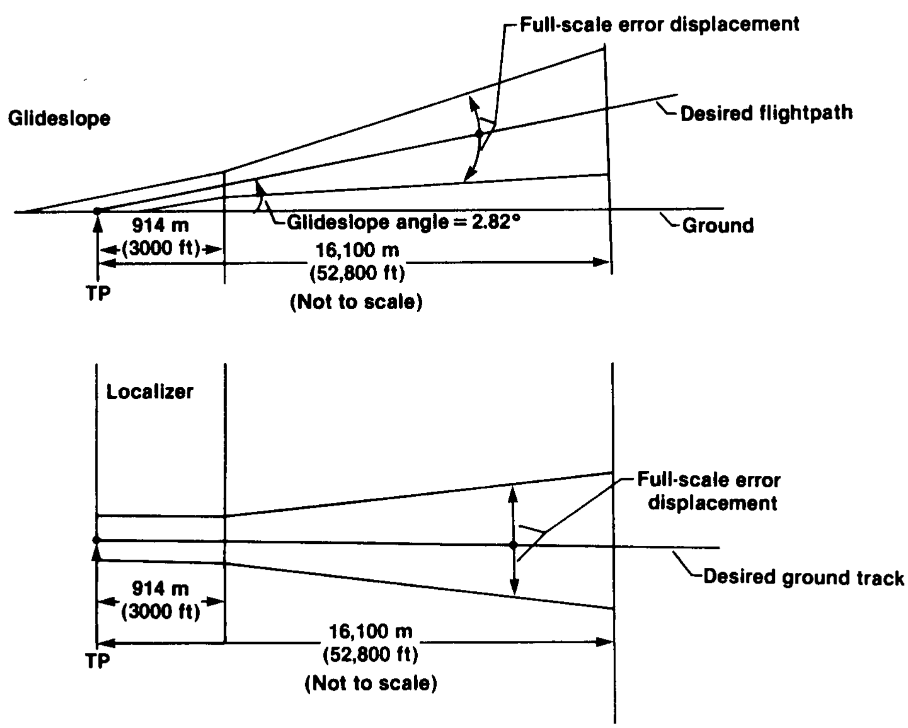


Figure 15. Glideslope and localizer.

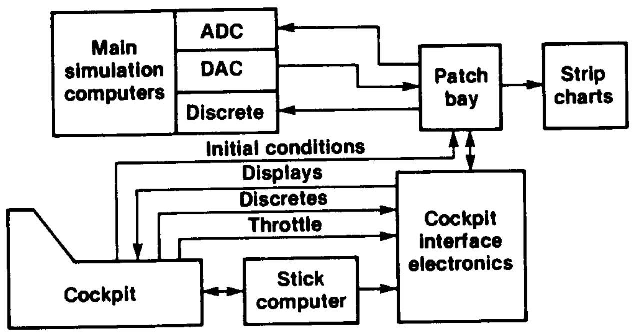


Figure 16. BASIC simulation.

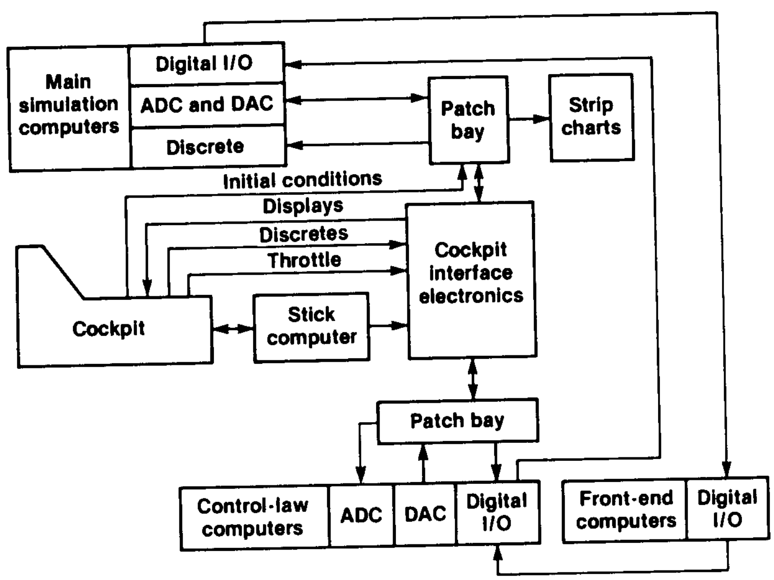


Figure 17. VERIFICATION simulation.

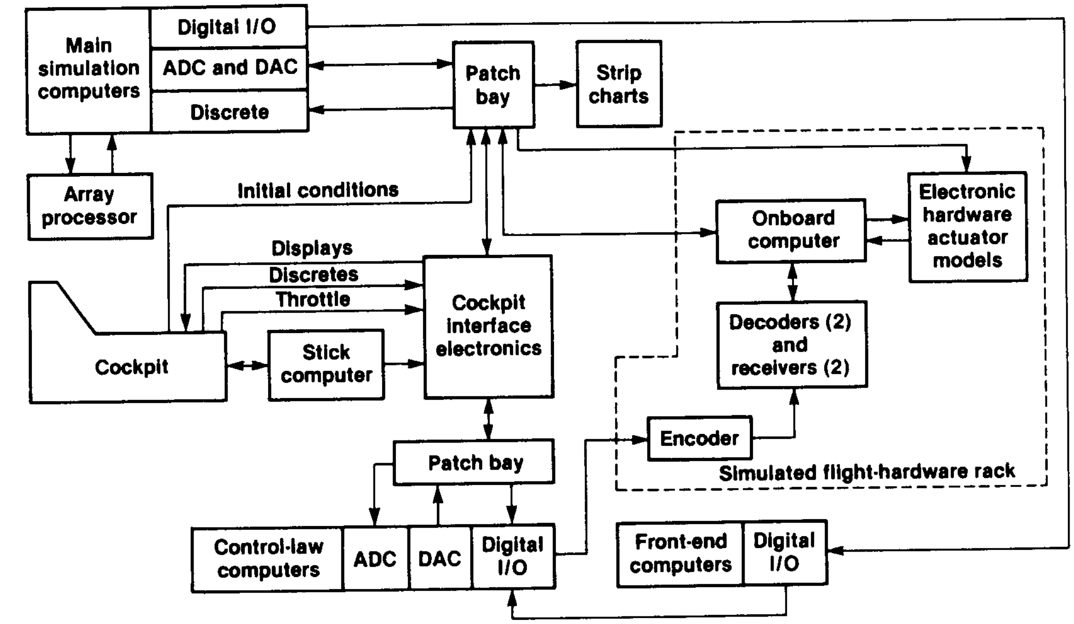


Figure 18. CASH simulation.

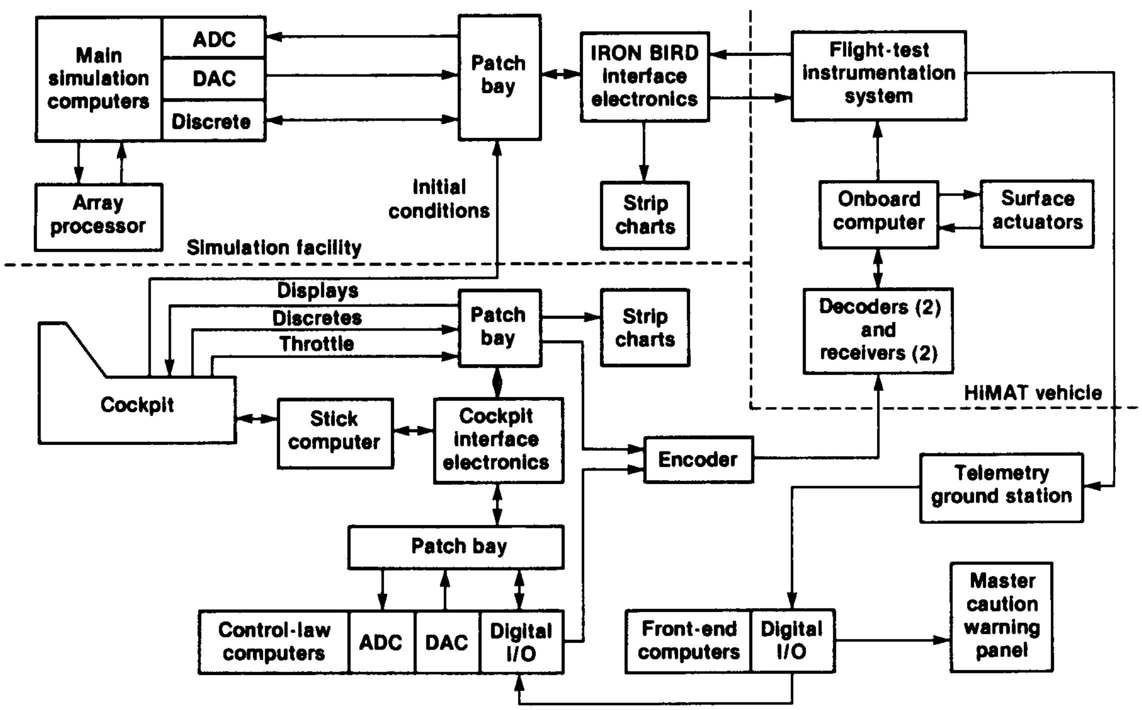


Figure 19. IRON BIRD simulation.

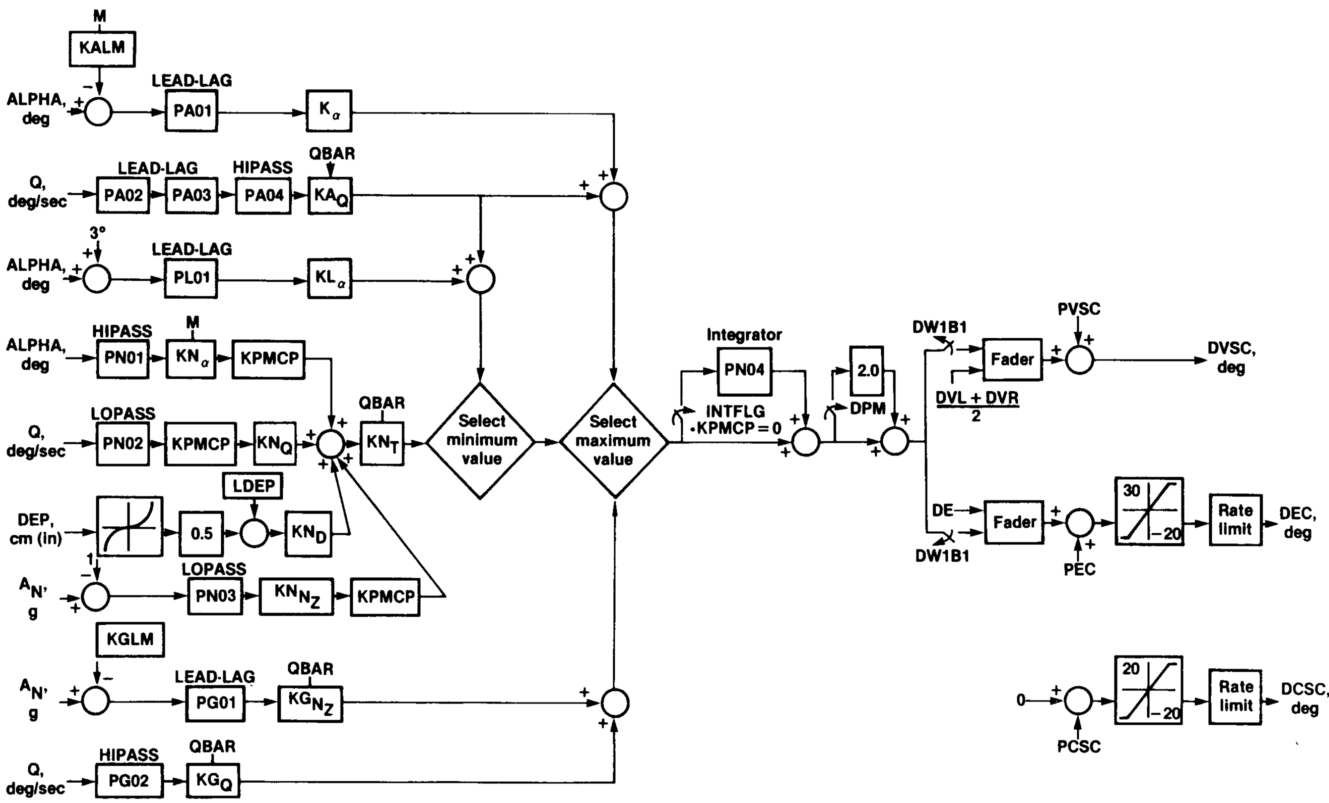


Figure B-1. Pitch axis control laws (PCS).

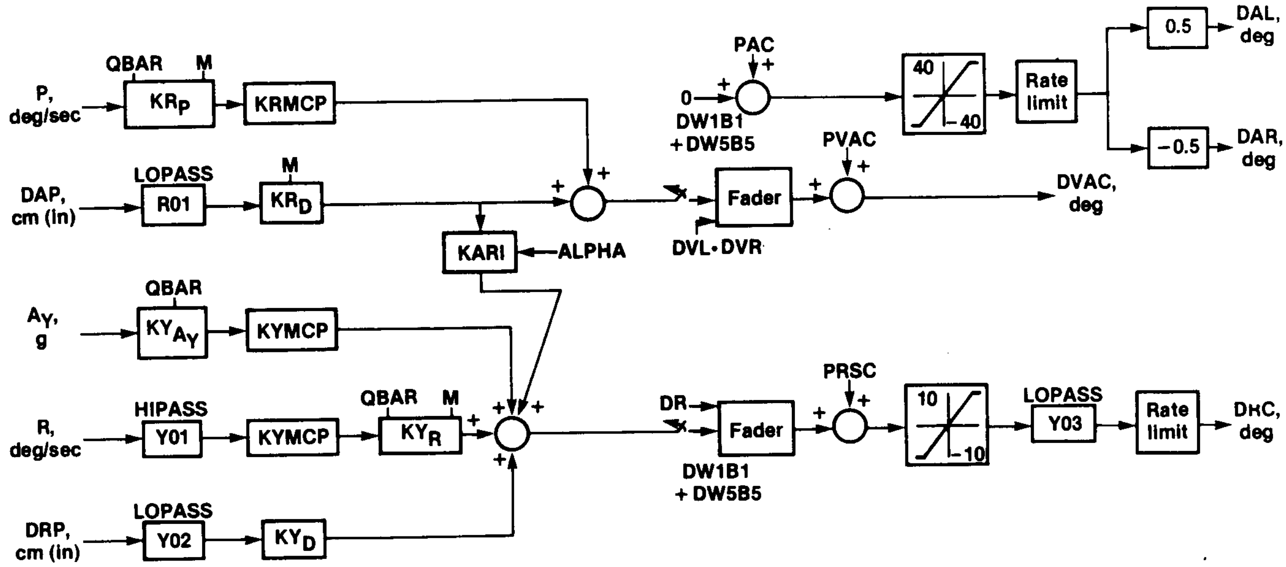


Figure B-2. Roll and yaw axes control laws (PCS).

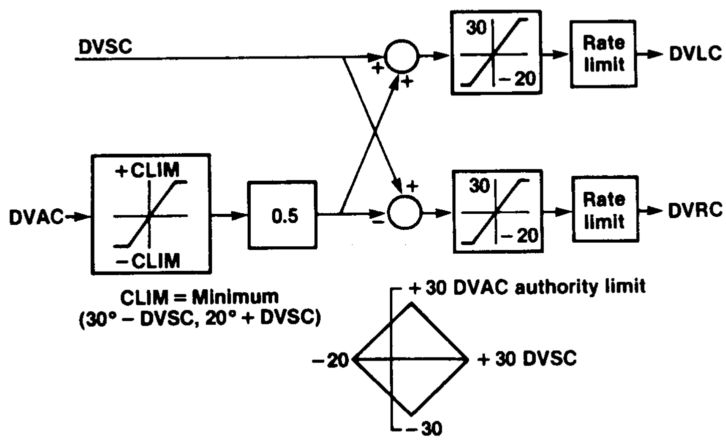
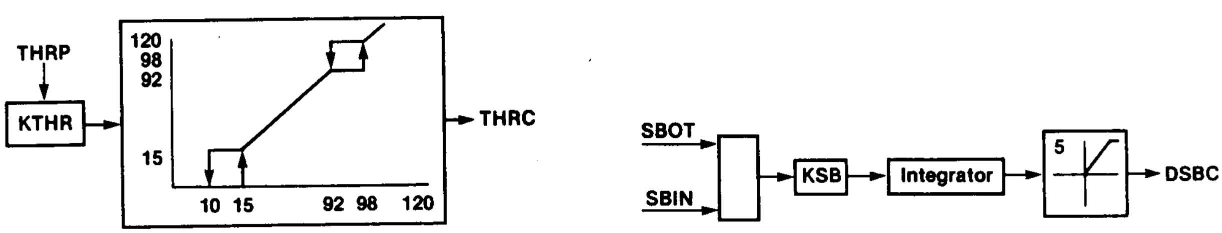


Figure B-3. Left and right elevon commands (PCS).



(a) Throttle. *(b) Speed brake.*

Figure B-4. Throttle and speed brake commands (PCS).

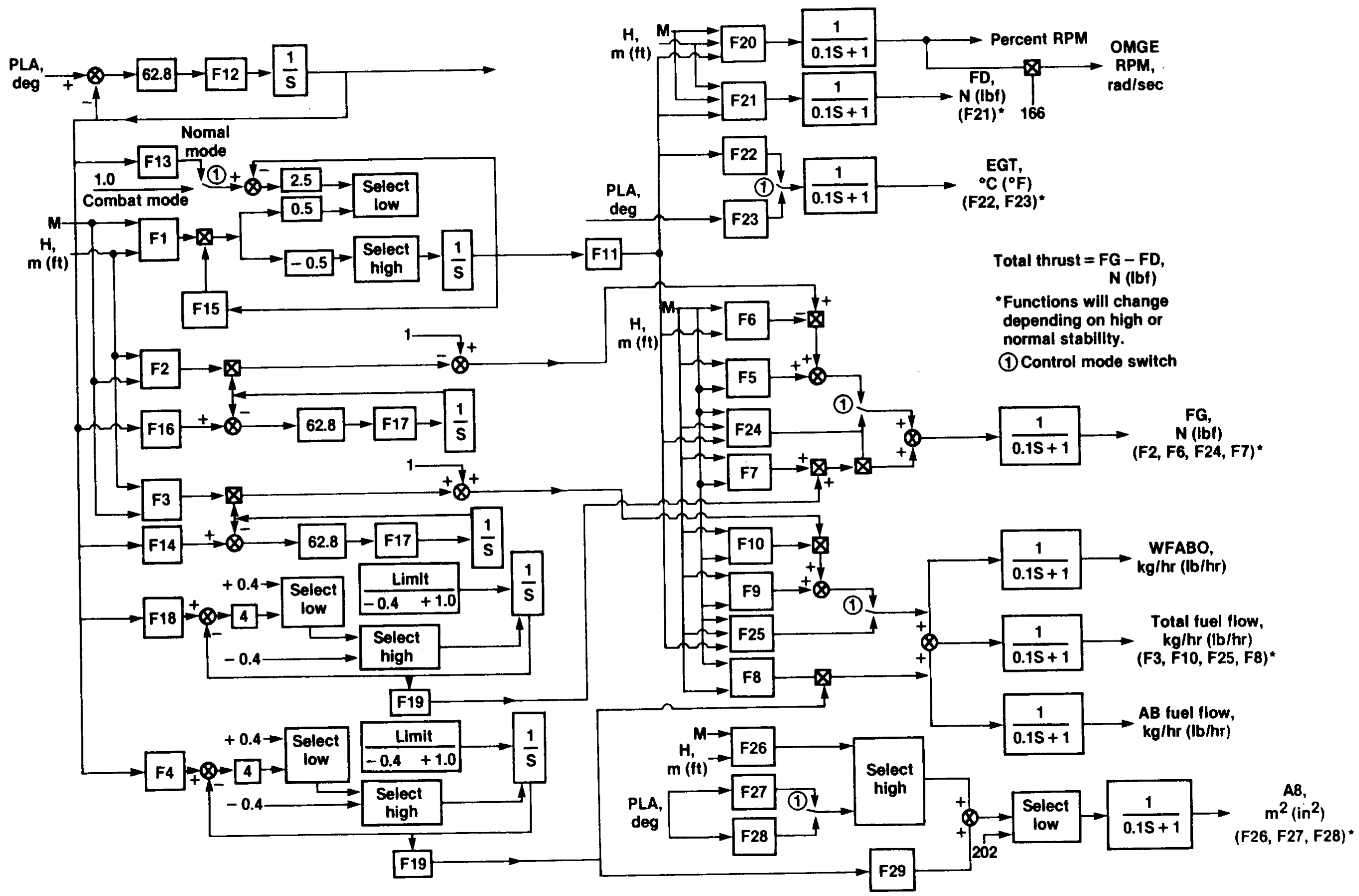


Figure C-1. HiMAT J-85 engine block diagram.

1. Report No.	2. Government Accession No.	3. Recipient's Catalog No.
NASA TM-84912		

4. Title and Subtitle	5. Report Date
The Role of Simulation in the Development and Flight Test of the HiMAT Vehicle	April 1984
	6. Performing Organization Code
7. Author(s)	8. Performing Organization Report No.
Martha B. Evans and Lawrence J. Schilling	H-1190
	10. Work Unit No.
9. Performing Organization Name and Address	11. Contract or Grant No.
NASA Ames Research Center Dryden Flight Research Facility P.O. Box 273 Edwards, California 93523	13. Type of Report and Period Covered Technical Memorandum
12. Sponsoring Agency Name and Address	14. Sponsoring Agency Code
National Aeronautics and Space Administration Washington, D.C. 20546	RTOP 533-03-11

15. Supplementary Notes

16. Abstract

Four real-time simulations were created for the highly maneuverable aircraft technology (HiMAT) program conducted at NASA Ames Research Center's Dryden Flight Research Facility. This paper details the software models, hardware configurations, and uses of each of the four HiMAT simulations.

17. Key Words (Suggested by Author(s))	18. Distribution Statement
Simulation Highly maneuverable aircraft technology (HiMAT) Simulation in flight test Remotely piloted vehicle (RPV) flight test	Unclassified-Unlimited STAR category 05

19. Security Classif. (of this report)	20. Security Classif. (of this page)	21. No. of Pages	22. Price*
Unclassified	Unclassified	37	A03

*For sale by the National Technical Information Service, Springfield, Virginia 22161.